United States Patent Application

for

CONTROL AND MONETIZATION OF NETWORKING TRANSACTIONS

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part and claims the benefit under 35 U.S.C. sec. 120 of Chudnovsky et al.'s co-pending United States Utility Patent Application No. 10/619,948, filed July 15, 2003, which prior-filed application claims the benefit under 35 U.S.C. sec. 119(e) of Chudnovky's United States Provisional Patent Application No. _____, filed July 1, 2003, both prior-filed applications being incorporated herein by reference.

TECHNICAL FIELD

The present invention relates generally to computer data and information systems, and more particularly to the control and monetization of networking transactions based on relational patterns between users.

BACKGROUND

The development of technology and the explosion of users connected to the Internet has permanently changed the way people communicate with each other. Networking databases can provide a user access to direct contacts to those family, friends and acquaintances the user has successfully added. A direct contact may be described as a contact separated from the user by one degree of freedom. Today it is possible to imagine a scenario where that user can access not just direct contacts, but contacts of direct contacts, i.e., other Internet users who are separated by more than one degree of freedom.

Tools exist for establishing relational or contact paths enabling a user to connect to other parties separated by more than one degree of freedom. In certain cases, these teachings provide the user a specific indication of the relational path that connects the user to the party.

Some of these tools have even provided unsophisticated techniques for monetizing transactions between users.

Existing technology fails to fully take advantage of the wealth of information available or potentially available through a properly populated and managed networking database. For example, in prior art schemes where relational connections are established between users, all relational connections are treated equally without reference to the degree of separation and other characteristics of the relational pattern. Likewise, there is no teaching for monetizing transactions between two users as a function of their degree of separation. What are needed are networking techniques and systems which fully utilize the relational data to implement sophisticated monetization and control strategies.

BRIEF DESCRIPTION OF THE FIGURES

The accompanying figures illustrate embodiments of the claimed invention. In the figures:

FIG. 1 is a block diagram of a networking database data structure in accordance with one embodiment of the present invention;

FIG. 2 is a block diagram of a user element data structure in accordance with another embodiment of the present invention;

FIG. 3 is a block diagram of a connections data structure in accordance with yet another embodiment of the present invention;

FIG. 4 is a graphical representation of relational connections for a networking database according to one aspect of the present invention;

FIG. 5 is a populated connections database in accordance with one embodiment of the present invention;

FIG. 6 is a graphical representation of relational connections for a second networking database according to one aspect of the present invention;

FIG. 7 is a populated connections database for the second networking database of FIG. 6 in accordance with one embodiment of the present invention;

FIG. 8 is a flow chart showing a first method for controlling and monetizing transactions according to another aspect of the present invention;

FIG. 9 is a flow chart showing a method for adding parties to a networking database in accordance with one embodiment of the present invention;

FIG. 10 is a flow chart illustrating a method for creating a relational connection having a type attribute between two users of a networking database according to one aspect of the present invention;

FIG. 11 is a flow chart illustrating another method for controlling and monetizing transactions among users of a database according to still another aspect of the present invention; and

FIG. 12 is a flow chart illustrating yet another method for controlling and monetizing transactions among computer users in accordance with yet another embodiment of the present invention.

Any headings used herein are for convenience only and do not affect the scope or meaning of the claimed invention.

SUMMARY OF THE INVENTION

The present invention teaches a variety of techniques and mechanisms for controlling and monetizing networking transactions among users of a computer network. At least some of these techniques and mechanisms come about as a result of several new paradigms contemplated by the present invention. One paradigm of the present invention involves monetizing user transactions as a function of relational patterns and connections between different users. Another paradigm of the present invention introduces a "virtual network" as a set of user having some predefined relational pattern. Once virtual networks are defined for a plurality of users, decisions regarding transactions can then be made based on the characteristics of the virtual network.

DETAILED DESCRIPTION OF THE ILLUSTRATED EMBODIMENTS

The present invention teaches a variety of techniques and mechanisms for controlling and monetizing networking transactions among users of a computer network. At least some of these techniques and mechanisms come about as a result of several new paradigms contemplated by the present invention. One paradigm of the present invention involves monetizing user transactions as a function of relational patterns and connections between different users. Another paradigm of the present invention introduces a "virtual network" as a set of user having some predefined relational pattern. Once virtual networks are defined for a plurality of users, decisions regarding transactions can then be made based on the characteristics of the virtual network.

The present invention defines a "networking transaction" broadly as any interaction which a unique computer user might contemplate performing. Thus networking transactions include communications of all types with other computer users, joining a networking database, inviting others to join a networking database, adding other users or parties to a users set of friends or contacts database, forming relational connections with other users, joining a community of other computer users having some common characteristic, creating a community open to other computer users having some common characteristic, inviting other users into a community, etc.

The computer network contemplated by the present invention can be a closed network, a wide area network, a local area network, a World Wide Web such as the Internet, etc. The present invention also teaches the use of networking databases populated with users and supporting viral growth. Such networking database are well suited for providing a structure wherein the control and monetization techniques of the present invention are implemented. However, the control and monetization techniques of the present invention can be accomplished with direct use of a networking database, as will be described below.

FIGS. 1 - 3 are block diagrams of several data structures suitable for use in managing networking databases of the present invention. In light of the present teaching, those skilled in the art will readily ascertain how to select specific implementations and variable types for the data structures of FIGS. 1 – 3. The networking database supported by FIG. 3 is a database populated with a plurality of uniquely identified users, together with data suitable to aid the control and monetization of requested transactions. Typically this data includes at least some of the following: user profile data, monetization data, relational information among the plurality of users, and transaction rules. Networking databases of the present invention enable control, implementation and monetization of transactions requested by the populating users. These transactions may involve parties that are not members of the networking database including those parties that have been invited to join the networking database.

FIG 1 illustrates a networking database 100 in accordance with one embodiment of the present invention. The networking database 100 includes a users data structure 102, a connections data structure 104, a paying users data structure 106, and an invited users data structure 108. The users data structure 102 is suitable for storing data characterizing and uniquely identifying the users populating the networking database 100.

The connections database structure 104 is suitable for storing data defining and/or characterizing relational connections between the users populating the networking database 100. Certain embodiments of the present invention teach that relational connections between users populating the networking database may have well-defined characteristics. For example, relational connection may have types such as "friends," "business," "family," "romantic interest," etc. Further, two users may have defined different relational connection types with

5

one another; for example a first user may define a relational connection with a second user as "friend," while the second user may define a relational connction with the first user as "business." Additionally, each relational connection may have several types, or it may be that two users have more than one relational connection. In any event, the nature and use of relational connection types will be discussed in more detail throughout the remainder of the specification.

With further reference to FIG. 1, the paying users data structure 106 is suitable for storing data related to user ability to monetize transactions. Data possibly found in paying users data structure 106 includes membership status (e.g., paying versus nonpaying), user subscription level, user account and credit information, etc. The invited users data structure 108 is suitable for storing data related to parties that have neither accepted nor rejected a pending invitation to join the networking database 100.

FIG. 2 illustrates a user element 103 suitable for use as a single element of an array of user elements forming a users data structure 102 in accordance with one embodiment of the present invention. The user element 103 includes user identification data 120, user email data 122, user first name data 124, user last name data 126, and user profile data 128. Once populated with the necessary data, the user element 103 corresponds to a unique specific user. The user identification data 120 is preferably a unique identification number selected by or assigned to the specific user. The intended contents of email data 122, first name data 124, last name data 126, will all be self-evident to those skilled in the art, and may be provided directly by the specific user or from any valid source. In certain embodiments, the user profile data 128 includes a privacy indication for data therein. The privacy indication can also be a function of the relational connection. E.g., only users having one degree of separation are entitled to search certain data, while the public may search other types of data.

Turning to FIG. 3, one embodiment of the present invention teaches a connections data structure 104 being an array having a column of user id 1 elements 140, a column of user id 2 elements 142, a column of connection status elements 144, and a column of connection type elements 146. When populated with data, a user id 1 element 140 contains a unique identifier for a first user present in the networking database 100, and a user id 2 element 142

6

contains a unique identifier for a second user present in the networking database 100. A corresponding connection status element 144 contains a status of a connection between the first and second users. The present invention contemplate a variety of connection status types including active, requested, uni-directional, conditional, etc. A corresponding connection type element 146 includes data defining the nature and/or type of the relational connection(s) between the first and second users.

FIGS. 4 and 6 are graphical illustrations of relational connections 202 for a specific networking database 200 in two different fixed states. FIGS. 5 and 7 show in table format connections data structure 103 for the fixed states of FIGS. 4 and 6 respectively.

Referring to FIGS. 4 and 5, the networking database 200 is populated with a plurality of users 1 – 16 having relational connections 202. FIG. 5 shows a populated connections data structure 103 representing the relational connections 202 of FIG. 4. As indicated by the status column, all the relational connections 202 are set at a status "A" representing that each relational connection 202 is active and available for supporting a transaction between the connected users.

Those skilled in the art will readily understand how certain relational connections can be viewed as a degree of separation between two users. By way of example, users 4 and 5 have two degrees of separation as they are connected by two relational connections 202 through user 1. In preferred embodiments, the relational path used to connect two users will be selected as the relational path with the least degrees of separation. Those users that have no connecting relational path have an undefined degree of separation.

In certain embodiments, the relational connections 202 are direct one degree of freedom relational connections between two users. However, those skilled in the art will appreciate that the relational connections 202 can take a variety of suitable forms. It is contemplated that a relational connection may correspond to a one way communication channel. For example, a first user may grant email access to all users in the networking database, however the first user could only send an email to those users that have granted the first user this right, either directly or indirectly through a series of suitable relational

connections 202. Furthermore, the present invention contemplates supporting different types of relational connections within the same networking database 100.

With further reference to FIG. 4, the users 1 - 16 are organized into virtual networks A, B, and C. A "virtual network" is defined as a set of all users within a networking database that have a relational connection or a defined degree of separation and are thus immediately available for certain transactions. The paradigm of virtual networks enables transaction control not previously available. Certain embodiments of the present invention contemplate limiting transactions the non-member parties can perform with members of a virtual network. This enables, e.g., elimination of spam email without resort to ineffective filters and other failed prior techniques. As will be appreciated, organization of users into virtual networks can be performed in real time during execution of transactions, or can be performed prior and the data stored within the networking database and updated periodically or upon any change. Virtual networks and their uses will be described in more detail below.

The virtual networks A, B, and C shown in FIG. 5 are of the type defined by a relational connection such as email contact, etc. However, virtual networks can be formed based on a variety of parameters such as those found in the user profile. For example, users may indicate their favorite movie genre, age of children, etc. Any of these profile parameters may be used to group users into virtual networks. This also enables users to be members of more than one different virtual network, where each virtual network formed based on a different type of user profile parameter.

FIG. 6 illustrates the networking database 200 after a state transition has occurred. Specifically, users 10 and 15 have established a relational connection 202, and users 9 and 10 have a requested relational connection 202'. The new relational connection 202 between users 10 and 15 causes virtual networks B and C to form a new virtual network BC. The requested relational connection 202' may have been initiated by either user 9 or user 10, or may have been initiated by the system. FIG. 7 shows in table format the connections data structure 103 for the fixed state of FIG. 6.

FIG. 8 is a flow chart showing a method 300 for controlling and monetizing a user requested transaction implemented within a networking database such as networking database

8

100 of FIG 1. Transactions contemplated by the present invention include communication (email, instant messaging, video conferencing, voice, etc.), database searching and viewing of public profile information, requests to add parties to the networking database or to a user's set of friends, etc. Each of these transactions are analyzed to determined whether monetization is desired. Monetization includes charging a user for a transaction involving another party, with possible reference to a variety of factors such as the relational connection of the user to the party, a membership or subscription level of the user and/or party, popularity of users, etc. Other types of monetization are described throughout the specification, and the intent here is for a broad all encompassing definition to apply. In light of the present teaching, those skilled in the art will readily implement still further mechanisms for monetizing transactions.

In an initial step 301, any necessary initialization and set up procedures are performed. The initialization process may involve instantiating a database manager process upon a database server. The initialization process may involve creating and initializing a template for a networking database, or invoking for use a pre-existing networking database. As will be appreciated, the networking database may be a distributed database, and the initial step would then involve establishing any necessary communications links between the distributed portions of the database.

In a step 302, a networking database is populated with a plurality of users. Populating the networking database with users may involve porting in a pre-existing database of users and inserting such data into the database template, updating the pre-existing database, inviting parties to join the networking database and then including received data as desired, responding to a party's request to be added to the networking database, creating a networking database from scratch, etc. Each user provides or is assigned a unique identifier, and a minimum amount of profile data is required. One suitable method 302' for adding a party to the networking database in response to a user request is described below in more detail with respect to FIG. 9.

A step 304 defines relational patterns among the users populating the networking database. Step 304 may include generating a connection table such as connection data structure 104 illustrated in FIGS. 3, 5 and 7, representing one degree of freedom connections

9

among the users populating the database. Step 304 may also define more complicated relational patterns. For example, the present invention contemplates one way connections between users, conditional or context sensitive connections between users, and connections of application and/or user defined type. For example, a user may set preferences such as limiting email transactions to only those users having a one degree of separation relational connection. As another example, a user may set a preference that email transactions must cost the requesting user.

Step 304 may acquire the data necessary to define and generate the relational patterns through any suitable mechansim or technique. For example, step 304 may mine through existing data found in sources such as contact databases, history of previous email traffic, etc. Step 304 may also include users entering email address information for contacts they have a relationship with. One suitable method for performing step 304 is described in more detail below with reference to FIG. 10

A next step 306 defines one or more virtual networks from the plurality of users populating the networking database. This step 306 is optional, and may be performed initially through examination of each user populating the networking database, or may be performed on a case by case basis as required to execute a transaction.

A step 308 defines a set of transaction rules for performing transactions. The present invention contemplates making a wide variety of transactions available to users in the networking database. These transactions include email, instant messaging, calendaring, data sharing, searching, auctioning, psychological profiling, dating services, community services. The set of transaction rules may include pre-existing rules taken from an outside source or invoked at initiation, or the transaction rules may be determined on the fly, and may depend upon the character of the networking database as populated. In certain embodiments, the transaction rules define how to monetize any transaction, and how to respond to requests originating from parties not found in the networking database, or transactions directed outside of the networking database.

A step 310 receives a transaction request. The received transaction request may originate from a user found in the networking database, or in certain embodiments a non-

member party. The present invention also contemplates absolutely closed networking databases that do not accept transaction requests whatsoever from outside the networking database.

A next step 312 processes the transaction request in a manner consistent with the transaction rules and the relational patterns of the networking database. Processing a transaction request may include determining whether the requested transaction is allowed and under what circumstances, determining the presence of required circumstances such as an existing relational connection which supports the requested transaction, requiring successful monetization prior to execution or initiation of such transaction, and/or attempting to change the circumstances in order to effectuate the transaction. One possible set of rules is illustrated below within the description of method 400 of FIG. 10.

A step 314 performs any necessary record keeping resulting from the transaction request such as updating the networking database and virtual networks, as well as notifiying a database manager of certain events.

The method 300 of FIG.8 is a suitable and generic approach to implement certain aspects of the present invention. However, those skilled in the art will readily be able to adapt the principles set forth in method 300 to more narrow applications as well as to broader embodiments.

FIG. 9 is a flow chart illustrating a method 302' suitable for responding to a specific user requesting that a party be added to the specific user's set of friends. A user's "set of friends" is defined as users found in a networking database that have a certain relational connection with the specific user. The present invention contemplates that the certain relational connection can be defined as any type or range of relational connections. For example, the certain relational connection may be defined as a direct, one degree of freedom connection. This direct connection implies that the two users are capable of performing any variety of transactions. In another embodiment, the certain relational connection may be a one way connection meaning that the specific user's friends have permission to send communications to the user, regardless of whether the specific user has permission to send communications to those users.

Allowing database users to add parties to their set of friends enables viral growth of the networking database. This viral growth in turn improves opportunity for transactions, and creates more opportunities for monetization of such transactions. Of course, certain embodiments of the present invention which require stricter control and privacy within the networking database will not enable or will strictly limit viral growth.

The method 302', as with any method for allowing a user to add members to the networking database, may result from the network database managing process proactively inviting the user to add to the networking database, say for example during an initial population phase of the networking database, or when the user has recently joined the networking database. The method 302' may be prompted by a determination that a requested transaction is not available until certain relational connections are established. Of course, the user may initiate a method 302' during the course of updating user contacts. Those skilled in the art will recognize that these are only a few of the ways and reasons a user may attempt to add a party to their set of friends.

Turning directly to the method 302' of FIG. 9, a step 350 receives a request from a specific user to add a party to a networking database. As mentioned above, the request of step 350 may arise in a variety of circumstances. A step 352 determines whether the party is a member of the networking database. When step 352 determines the party is present in the networking database, a step 354 determines whether the party is willing and available for inclusion in the specific user's set of friends. Certain implementations of step 354 require explicit, case by case, permission from the party. In other embodiments, a predefined variable prohibiting connections or conditionally or unconditionally enabling connections can be set by the party, or by the managing process. For example, a party may only permit inclusion in the set of friends when the user has a certain level of popularity as defined by the system. In other embodiments, step 354 evaluates a status of the party, e.g., perhaps the party is a member of the database but is delinquent on payment for services, etc. Step 354 can be skipped altogether, for the present invention contemplates schemas where all users in the networking database are entitled to add other users into their set of friends.

12

When step 354 determines that the party either denies permission or is not entitled to become a member of the user's set of friends, a step 368 rejects the user's request. The rejection of step 368 often includes providing some type of feedback to the requesting user regarding the failure. In certain embodiments, step 368 includes a determination of whether the requesting user can accomplish the request by taking some other action such as adding another party to the database, and then prompting the user to take such action first and then resubmit the request. For example, the party may only grant permission if the user has a relational connection to another user found in the party's set of friends. In this case, the user may be prompted to add another user with such a connection thereby enabling addition of the desired party.

When step 354 determines that the party grants permission and is entitled, a step 356 monetizes the transaction. Monetization may include charging the requesting user for adding the party to the user's set of friends, or determining if the requesting user has a subscription level allowing the transaction (e.g., nonpaying users cannot add, paying users can). The present invention contemplates embodiments that involve sharing the benefit of the monetization with the added party, and/or allowing the added party to set the cost of the transaction. Alternatively, monetization may include giving credit to the requesting user, or may simply determine that the cost to the user and the party is nothing (i.e., free!).

When monetization step 356 fails, a step 368 rejects the user's request. When monetization failure causes the rejection, step 368 often includes providing feedback regarding the nature of failure. In certain embodiments, step 368 includes a determination of whether the requesting user can accomplish the request by taking some other action such as updating credit card information or providing another valid form of payment, taking an action which results in credit being added to the user's account, etc. Step 368 may prompt the user to attend to these and resubmit the request to add the party to the user's set of friends.

When monetization step 356 succeeds, a step 358 adds the party to the user's set of friends by taking an action such as updating the connections data structure. Then a step 360 provides feedback to the requesting user regarding the success of the transaction.

Turning back to an earlier part of the flow chart not yet covered, when step 352 determines that the party is not a member of the networking database, a step 362 determines whether the party can be added to the networking database. Step 362 can be based on a variety of factors such as the identity of the party (known spammer?), whether a suitable connection can be made with the party, etc.

When step 362 determines that the party is not allowed in the database, a step 368 rejects the user's request, typically providing feedback in the process. When step 362 determines that the party is eligible to join the networking database, a step 364 invites the party to join the networking database. The process of joining typically involves obtaining data such as that necessary to populate the networking database with the party, assigning a unique user identity, monetizing the transaction as desired, etc. When the party successfully joins the networking database, control passes to monetization step 356, and on to steps 358 and 360 as described above. When the party does not join the networking database, control passes to step 368 that rejects the user's request and provides feedback to the user as desired.

FIG. 10 is a flow chart illustrating one suitable method 304' for defining and adding a relational connection between a first user and a second user of a networking database. The method 304' implements the formation of one network connection and could be used iteratively to define the relational connections among all users of the networking database as a portion of the step 304 described above with reference to FIG. 8.

With further reference to FIG. 10, a step 370 receives a proposed relational connection between the first and second user of the networking database. The proposed relational connection may arise from any suitable source such as a pre-existing database, a specific request of the first user, or during an automated process executing to establish relational connections based on other available data. The proposed relational connection may include a status and a type as mentioned above.

A step 372 determines whether the proposed relational connection type is defined. For example, the networking database may have a set of predefined types such as "friend," "business," "family," etc. In some embodiments of the present invention, users of the networking database will be allowed to define desired relational connection types. When the

14

relational connection type is not already defined, control passes to a step 374 which enables definition of the new relational connection type.

In any event, a step 376 determines a status of the proposed relational connection. For example, the proposed status may be active, uni-directional, bi-directional, etc. A step 378 attempts to implement the proposed relational connection. In some instances, the second user may be queried for permission to establish the proposed relational connection, predefined settings may be evaluated, etc. A step 380 monetizes the establishment of the new relational connection. For example, certain users may only allow connections to be established based upon payment, or the system may charge for creation and maintenance of such connections.

A step 382 determines whether the proposed relational connection has been established. If not, an optional step 384 provides an error message to the user, or perhaps prompts the user to take additional steps to effectuate creation of the proposed relational connection. When the connection is established, a step 386 updates a relational connections data structure to reflect the new relational connection.

FIG. 11 is a flow chart illustrating another method 400 for controlling and monetizing transactions among users of a networking database. The method 400 can be interpreted as a partial or whole implementation of the set of transaction rules described above with reference to FIGS 8, or alternatively may be interpreted as a suitable stand alone method for controlling and monetizing transactions among users of a networking database. In either case, a preliminary step 401 corresponds to all the necessary initialization steps, such as populating the networking database, which set the stage for dealing with user transaction requests.

A step 402 receives a transaction request from a specific user. The present invention contemplates a wide variety of available transactions such as email, instant messaging, video conferencing, database searching, modifying the database, dating services, job finding services, people finding services, contact lookup, etc. The transaction request may also include a particular parameter further defining the requested transaction. For example, the specific user may request a family search for a second user, the return results would be a list of all users having a connection of type "family" with the second user. A step 404 determines the nature of the requested transaction. A step 406 determines whether the transaction is feasible based

15

on a set of predefined rules. For example, the first user may request communication with a second user, in which case it must first be determined whether there is a relational connection between the first and second user sufficient to support the requested communication. Perhaps the transaction request corresponds to a search on a particular type of user profile data, and step 406 must determine whether the first user has access to such data.

In certain embodiments, when step 406 determines the requested transaction is not feasible, the method 400 proceeds to a step 418 that performs any necessary updates to the database. For example, it may be useful in certain contexts to monitor failed transactions. Then a step 420 responds to the specific user indicating that the requested transaction failed, and if desired, the response can indicate the nature of the failure.

In other embodiments, when step 406 determines that the requested transaction is not feasible, the method 400 may continue in step 406 to attempt to change the situation so that the transaction is feasible. For example, email may only be permitted among users that are members of the same virtual network. Thus a request for communication to a user outside of the specific user's network is not immediately feasible, but would be feasible should the specific user add a relational connection that includes the outside user into the same virtual network. Step 406 may prompt the specific user to take action to accomplish this, and then hold the transaction request for a period of time if during which the user succeeds in adding the outside user to the specific user's virtual network, the transaction becomes feasible and the method 400 continues rather than terminating the transaction request.

In any event, when step 406 determines that the transaction is feasible, a step 408 then determines whether the transaction is free. When the transaction is free, control passes to a step 416 which initiates the requested transaction, then a step 418 performs any necessary database updates, and then step 420 provides a response to the specific user regarding the successful initiation of the transaction. The present invention contemplates that the user may be compensated for successfully initiating certain types of transactions. For example, the specific user may be compensated for transactions resulting in another paying user joining the networking database, and/or transactions such as completing surveys, or joining another networking database, or joining a given virtual network. This compensation may take a

16

variety of forms such as direct payment, credit for future transactions within the networking database, credit toward the purchase of goods sold by the operator of the networking database, airline miles, etc.

When step 408 determines that the transaction is not free, a step 410 determines whether the specific user is eligible to have the requested transaction completed based on a subscription level or some other characteristic of the specific user. An example rule might instruct us that email communication between users having one degree of separation is free and becomes progressively more expensive as the degree of separation between users increases. When step 410 determines that the specific user is eligible, process control passes along to step 416 that initiates execution of the requested transaction and proceeds as described above.

When step 410 determines that payment is required, a step 412 requests payment from the specific user. It will be appreciated that payment can be accomplished through a variety of mechanisms. For example, the specific user may have an account with the networking database, and step 412 debits the transaction cost from that account if funds are available. The specific user's account may have both a cash reserve, credit from past monetize transactions, or both. Certain transactions may be payable only from a cash reserve, while others may be payable from cash reserve or credit. As another example, the specific user may have a credit card on file that is automatically debited or the user may be prompted to select a credit card to pay the transactions cost.

A step 414 determines whether the transaction cost was successfully paid, and if so control passes to transaction execution step 416 and proceeds as described above with reference to initiating and executing the transaction, else the transaction request is rejected and the method 400 proceeds as described above with reference to failure.

FIG. 12 illustrates another method 500 for controlling and monetizing a transaction between two users. The above discussion focused on networking transactions based on a specific networking database. However, the teachings of the present invention are not limited to networking database scenarios, but rather can be applied to any transaction between users

17

found on a computer network. In the example of FIG. 12, method 500 focuses on monetizing transactions between two users as a function of a relational connection between the two users.

A first step 502 receives a request from a first user to perform a transaction that is dependent upon a second user. The requested transaction can be any of a variety of suitable transactions such as communication (email, etc.) with the second user, a search upon data related to the second user, etc. A next step 504 determines a relational connection between the first and second users. Step 504 contemplates both real time determination and retrieval of the relational connection information from a networking or other type of database.

After the relational connection is determined in step 504, a step 506 determines a cost of executing the transaction, the execution cost being a function of the relational connection. The method 500 does not limit how this cost is determined except that it must in some way be related to the relational connection between the two users. For example, it may that transactions between users having a one degree of separation relational connection are free. The transaction cost may increase as a function o the degrees of freedom of the relational connection. Or perhaps all email transactions are free between users that have a relational connection, but video conferencing transactions and searches are monetized as a function of the relational connection between the two users.

In any event, once the cost has been determined, the method 500 continues as a step 508 performs initiation or execution of the transaction if and when the costs are met. As described above, the cost may be met in a variety of ways, and the benefit may be shared with the second user. Certain transaction may result in benefit flowing to the first user, which benefit may take on any suitable form.

The description herein of illustrated embodiments of the invention is not intended to be exhaustive or to limit the invention to the precise form disclosed. While specific embodiments of, and examples for, the invention are described herein for illustrative purposes, various equivalent modifications are possible within the scope of the invention, as those skilled in the relevant art will recognize.

Access to the different forms of databases and virtual networks can be controlled by any means. For example, access to a virtual network may be controlled such that only members of the virtual network may search on certain profile data associated with members of the virtual network. Alternatively, access to a virtual network may be controlled so that non-members may search or view certain profile data associated with members of the virtual network, but non-members are not allowed to engage in transactions with the members of the virtual network.

Regardless of any serial execution of the flow chart steps implied by their visual placement, those skilled in the art will recognize when such steps may be performed in parallel or in a different order, sometimes without effecting the outcome and other times in order to implement a different but related embodiment.

All of the references and U.S. patents and applications referenced herein are incorporated herein by reference. Aspects of the invention can be modified, if necessary, to employ the systems, functions and concepts of the various patents and applications described herein to provide yet further embodiments of the invention.

These and other changes can be made to the invention in light of the detailed description herein. In general, in the following claims, the terms used should not be construed to limit the invention to the specific embodiments disclosed in the specification and the claims. Accordingly, the invention is not limited by the disclosure, but instead the scope of the invention is to be determined entirely by the claims.

While certain aspects of the invention are presented below in certain claim forms, the inventors contemplate the various aspects of the invention in any number of claim forms. Accordingly, the inventors reserve the right to add additional claims after filing the application to pursue such additional claim forms for other aspects of the invention.